sciendo

Transport and Telecommunication, 2023, volume 24, no. 3, 256–265
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia
DOI 10.2478/ttj-2023-0020

TRAFFIC MONITORING SYSTEM FOR VEHICLE DETECTION IN DAY AND NIGHT CONDITIONS

Ibtissam Slimani[1], Abdelmoghit Zaarane[2], Issam Atouf[3]

[1,2,3]LTI lab, Faculty of Sciences Ben M'sik
Hassan II-Casablanca University
Casablanca, Morocco
[1]ibtissamslimani7@gmail.com

In this work, a day and night time vehicle detection system for traffic surveillance is proposed. Our system is composed of two main processes, day time and night time processes. In the night time, the vehicles are detected based on their taillights and headlights. First of all, the 2D-DWT (Two Dimensional Discrete Wavelet Transform) and the background subtraction are applied to the input image. Then, the connected component technique is used to extract the region of interest. If it is the daytime, the connected component candidates are taken as potential vehicles after applying a pre-processing algorithm to improve the result. If it is the night-time, a filtering operation is used to keep only the bright white and red connected component candidates (which represent potential headlights and taillights, respectively). Finally, potential lamp sets are formed by grouping the extracted components on the basis of their positions, sizes, and colours. The potential extracted vehicles are classified as a vehicle or non-vehicle by using a pre-trained CNN (Convolutional Neural Network) classifier. The proposed system was tested and evaluated using different works from the literature. The experiments show that our proposed system has reached a high accuracy in terms of vehicle detection process whether in day or night time. The experiments were performed using four different videos and were implemented using the C++ language, which facilitates mathematical computation, and its OpenCV library, which is used to run the image processing algorithms used, as well as the TensorFlow library, which facilitates transfer learning of pre-trained CNN models.

Keywords: Image processing, Traffic surveillance system, Vehicle detection, Day-time detection, Night-time detection, Discrete wavelet transform (DWT), Convolutional neural network (CNN)

1. Introduction

In Intelligent Transportation Systems (ITS), traffic monitoring is a major issue. The safety of people and property is at real risk as road accidents increase year on year and as a result traffic monitoring systems (Slimani *et al.*, 2020; Slimani *et al.*, 2019) and driver assistance systems (Zaarane *et al.*, 2020; Zaarane *et al.*, 2019b) have emerged and developed. Nowadays, vehicle detection (Al Okaishi *et al.*, 2019; Zaarane *et al.*, 2019a) is a very important subject because of its major role in reducing accidents and controlling traffic, especially on freeways. Different types of sensors are employed for detecting vehicles, e.g. laser scanners, radars and ultrasound sensors, but they are unable to detect multiple vehicles and cannot distinguish between vehicles and different obstacles (cars, pedestrians ....). This is why, recently, researches in the fields of surveillance and intelligent vehicles have focused and shown more interest in computer vision-based detection technologies. However, so far, all the solutions provided by these researches are still insufficient to meet the needs of road safety. For this reason, our work is oriented towards this line of research.

In this work, day and night vehicles detection system based on image processing is proposed. This system aims to detect vehicle during day and night in real time to extract the traffic condition information and controls the state of road traffic. Several methods are proposed to detect vehicles during day and night time.

Most researchers focus on vehicle detection during daytime because of the good lighting conditions. Based on an enhanced YOLO v5 network (You Only Look Once Version 5), the authors in (Zhang *et al.*, 2022) provide an enhanced approach for vehicle detection in various traffic circumstances. The flip-mosaic technique is used in the suggested approach to improve the perception of the network for small targets. You Only Look Once (YOLO) is a popular family of object recognition models, first described by (Redmon *et al.*, 2016). The approach involves a neural network that puts a multitude of neurons together, takes a photo as input and directly predicts bounding boxes, more commonly known as "bounding boxes", and class labels for each bounding box. The input image is initially divided into cells within this model, each of which is used to predict a bounding box. A class prediction is also based on each cell.

In (Sudha *et al.,* 2020), the authors suggest an Enhanced You Only Look Once (EYOLO) v3 deep learning technique to identify vehicles during the day and depict a square bounding box using a variety of labels such as car, bus, truck and heavy truck. In addition, they suggest a method known as Combined Filtering Techniques and Multiple Vehicle Tracking Algorithms (MVTAs) to update the region predictions and reset the size of windows, correlation average and covariance to detect the trajectory of vehicles in the next sequences of frames.

The authors in (Velazquez-Pupo *et al.,* 2018) present a daytime vehicle detection system with a single static camera for traffic surveillance based on occlusion handling, tracking, counting, and One-Class Support Vector Machine (OC-SVM) classification. (Billones *et al.,* 2017) present a vehicle detection and tracking system based on corner feature point detection and artificial neural networks for the vision-based contactless traffic violations apprehension system.

In (Tang *et al.,* 2017) the authors present a vehicle detection and recognition system based on the characteristics of the vehicle images for vehicle detection in road scene images. Their proposed system is based on Haar-like features and AdaBoost algorithms to extract features and construct classifiers.

However, the use of a day-time detection system to detect a vehicle at night never provided significant results because of the weak illumination conditions. For this reason, there are some proposed methods that are specified for the night time detection.

In low-light conditions, light sources like headlights and taillights are the elements that stand out the most. Vehicle lights have evolved into useful elements that may be used to distinguish vehicles. A vehicle detection algorithm can be designed using either taillights, headlights or both, depending on the objective of the individual system.

In (Gao *et al.,* 2019) authors propose a night-time vehicle detection method based on saliency detection for traffic surveillance system. They first use the frame difference technique to identify the objects in motion, before extracting all of the vehicle's tail lights using the saliency map and colour details. They then use all the tail lights in tandem to find the moving vehicles.

Similarly, the authors of (Vu *et al.,* 2017) propose a night vehicle detection as well as classification system for traffic monitoring. Their method includes headlamp segmentation, headlamp recognition, headlamp tracking and matching, and vehicles classification. First, by exploiting the variations in the colour and the luminance, they segment the bright objects. Then they use headlamp characteristics like centroid, area and shape to detect the candidate headlamps. Finally, they suggest a method to track and pair headlamps based on computing the area ratio.

Bell *et al*. (2021) have developed an algorithm for detecting vehicles in night-time conditions by analysing complex light patterns. They created a new machine-learning framework that utilizes a grid of foveal classifiers, which focus on specific areas of the image. This approach allows for more precise detection, but is limited to the areas targeted by the foveal classifiers. Instead of using bounding boxes, this algorithm utilizes point-based annotations to locate the detected vehicles.

In (Miao *et al.,* 2020), the authors propose a night vehicle detection method based on YOLO v3. The process begins by utilizing an optimal MSR algorithm (Multi Scale Retinex algorithme) to improve the brightness and clarity of original night-time images. Subsequently, a pre-trained YOLO v3 network is fine-tuned using a training dataset derived from the enhanced images. Following multiple training rounds, the refined model is employed to extract features from the images. Ultimately, the network is utilized to detect vehicles in the test dataset by identifying their locations and classifications.

In this regard, the goal of our work is to design an efficient, fast and low-cost vehicle detection system based on computer vision, which detects vehicles during day and night in real time with high accuracy. To do so, we have relied on deep learning in addition to various simple, powerful and easy to implement image-processing algorithms.

In this paper, we present a vehicle detection system that utilizes computer vision techniques, which detects vehicles during day and night in real time. This system is composed of five main parts. The first part is the pre-processing part, which is used to denoise and compress the input images. The second part is the road detection part, which is used to limit the processing area. This is important because limiting the area of interest leads to a reduction in processing time. This process is only applied once, when the system is powered on, which means that it will not have any negative impact on the processing time. The third part is the background update part, which is used to handle the differences that occur in the scene from the background image over time due to changes in weather and lighting conditions. The fourth part is the part of detecting and extracting areas of interest that may be vehicles. The potential vehicles are extracted and generated based on the background subtraction technique. These potential vehicles are then verified and classified using CNN in the last part. Figure 1 show the overall diagram of our proposed work.

The structure of the paper is arranged as follows. The beginning of the paper, Section I, provides an overview of the topic. The vehicle detection method is outlined in Section II. The experiment results are detailed in Section III, and the paper concludes in Section IV.

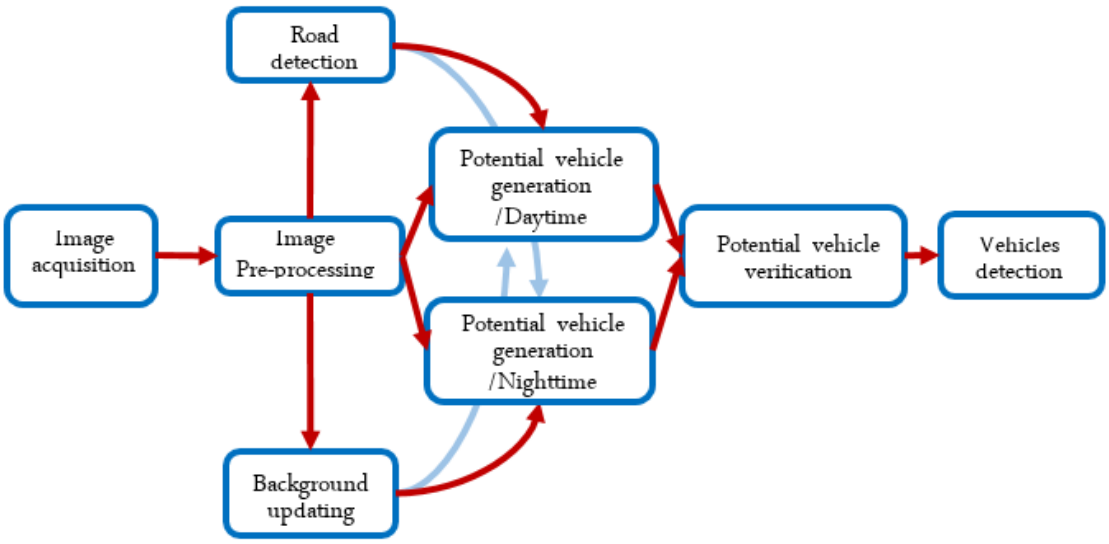


Figure 1. Overall flowchart of the proposed vehicle detection system

2. Vehicle detection method

The aim of the proposed method is to address the issues of daytime and night-time vehicle detection and to improve the results of our previous algorithms, the principle of which is described in this section. The proposed vehicle detection system is adapted to detect vehicles by video stream from a static camera mounted on a roadside pole for traffic monitoring.

The organization of this section is divided into: candidate generation which is a common step in both detection modes, vehicle candidate verification during the day and lamp candidate verification during the night.

2.1. Candidate generation during day/night time

The candidate generation step is an important step in the vehicle detection system, where the quality of candidate generation directly affects the accuracy of detection. Therefore, we perform several algorithms to properly generate candidate vehicles based on our previous work (Slimani *et al.*, 2018) by first performing the 2D-DWT **(**Two Dimensional Discrete Wavelet Transform**)** to extract the useful features. Then, we apply background subtraction to extract the foreground objects and going through updating the background image each period to improve the quality of the background subtraction. And finally, we perform the technique of connected components to extract regions of interest (vehicle candidates).

- *Methodology*

First of all, we perform on the input images a pre-processing and decomposition using 2D-DWT. Where the 2D-DWT (Slimani *et al.*, 2016) is a technique that decomposes the input image into four sub-images, three high-frequency sub-images, and one low-frequency sub-image. Where the low-frequency sub-image is the image that contains the most important information about the input image. In order to reduce noise and computational costs, the low-frequency sub-images of the second level of the 2D-DWT are the accredited images on this system. The choice of the filters is depending on the choice of the mother wavelet where the Haar wavelet is used in our work.

After transforming the input images, the background subtraction method using Gaussian Mixture Model (Nurhadiyatna *et al.*, 2013) is used to detect areas of interest, which are vehicle candidates for detection during the daytime and lamps candidates for detection during the night. The principle of this method is to extract the new foreground objects in the input image by generating a binary image from a

subtraction between the input image and the background image. For the input images in day time a thresholding operation is performed on the results of the subtraction using the choice of a medium threshold to omit small and slight lighting changes and retain only large and sharp changes (Slimani *et al.*, 2018), see Figure 2. For the night time, the thresholding operation used is based on the choice of a high threshold to retain only those bright regions that would be the lamp candidates, see Figure 3.

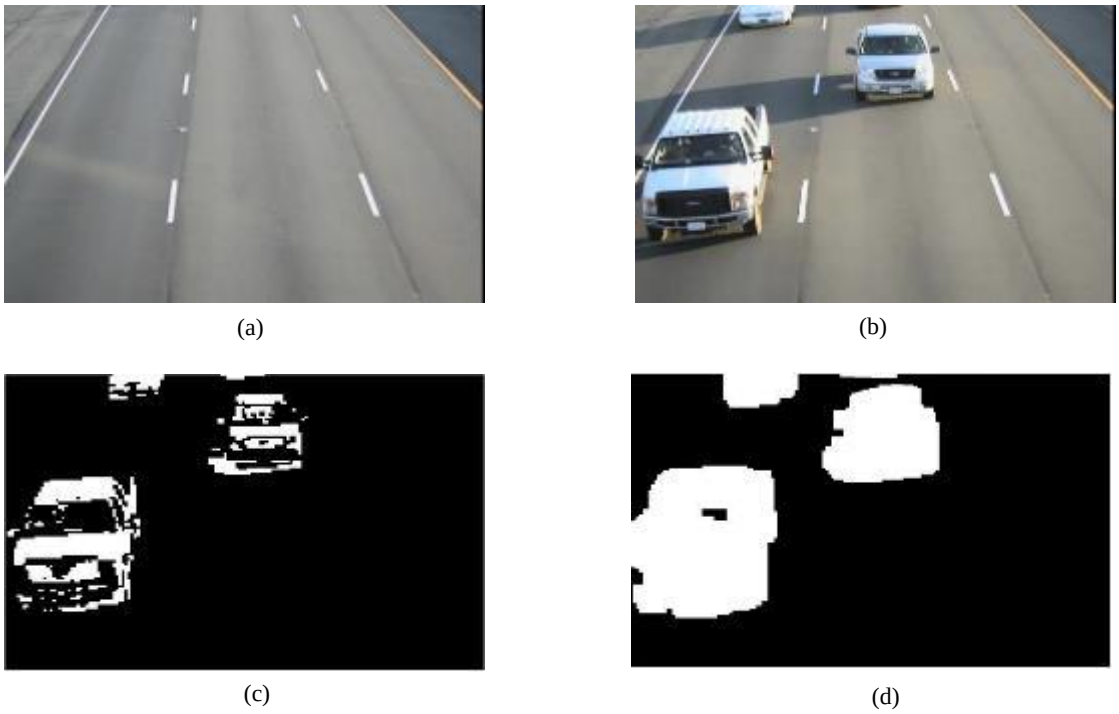


Figure 2. (Slimani *et al.*, 2018) Example of background subtraction for daytime images: (a) Extracted image of background; (b) input image; (c) Subtraction between (a) and (b); (d) Image of the foreground after the reconstruction

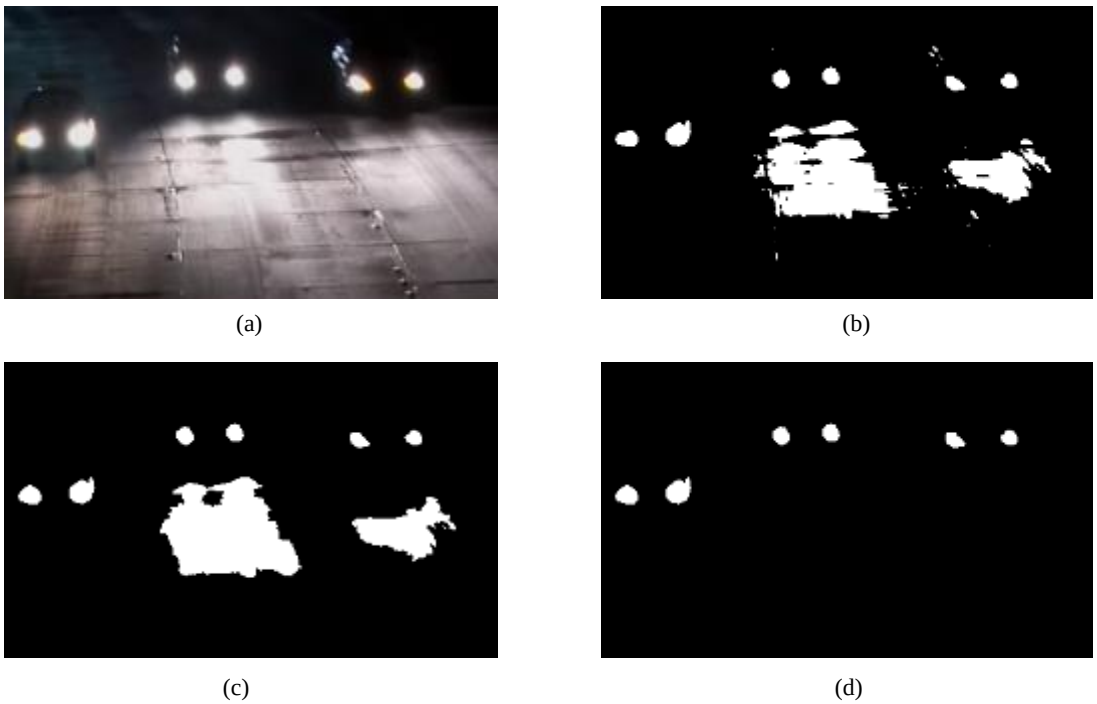


Figure 3. Example of background subtraction for night-time images: (a) input image; (b) Image of the foreground; (c) Image of the foreground after the reconstruction; (d) Potential lamps extracted

In order to improve the results of background subtraction, we periodically update the background image, as proposed in our previous work (Slimani *et al.*, 2018). Due to the difficulty of updating the entire image in a single shot, especially during rush hours, we update the background image line by line by measuring the degree of change in that line. Specifically, we update the lines that have been slightly modified by subtracting the lines at each period and updating only the line whose subtraction is less than a predefined threshold. This means that each line in the active image is treated as an empty one, when the corresponding line in the binary foreground image has a number of white pixels below a predefined threshold. Therefore, the new line in the active image will replace the existing line in the background image, and its flag will be changed to 1 to specify that the active line has been updated. Thus, it will remain unupdated until its flag is cleared. Otherwise, if the line of the active image is considered as not empty, the old line of the background image would not be updated. And it will be checked in the next input images until it is updated. The indicators are cleared periodically every certain number of input images which is actually adjustable to update the background image (see Figure 4).

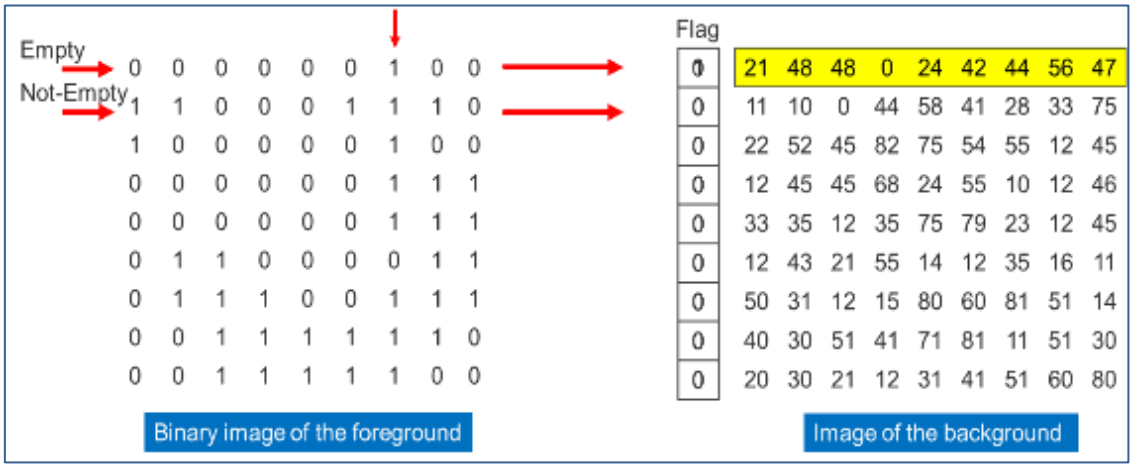


Figure 4. Background image updating

The binary image generated by background subtraction can contain small objects that cannot be candidates (areas of interest), which in our case are considered noise, and it can also contain objects with holes that can divide them into two or more separate objects. To solve this problem, we perform some morphological functions, dilation followed by erosion, to restore the holes and remove the noises. Then we extract each of the remaining candidates as a single connected component by performing the connected component technique (Gonzalez *et al.*, 1992), where this technique scans the binary image pixel by pixel, grouping together all adjacent pixels of each candidate to form a single connected component. This technique extracts the candidates by determining their coordinates and dimensions.

2.2. Candidate verification during day/night time

The candidate verification step is as important as the previous step, where the choice of the used classifier is a very important task that strongly affects the accuracy of the results, and where the right choice of the classifier significantly improves the accuracy rate. Therefore, we have used the CNN classifier in order to properly classify the candidates generated during day-time and night-time.

- *Candidate verification during day time*

In order to verify the candidates generated during the day in the previous step, we used the CNN classifier, where it is retrained to classify the generated candidates as vehicles or non-vehicles. The dataset used to retrain the CNN contains positive samples, which are vehicle images collected from several videos from several countries, and negative samples, which are various images containing different objects or parts of the road.

Convolutional neural networks (Nogueira *et al.*, 2017) are a type of artificial neural networks used frequently in image recognition and processing, and specially designed for pixel analysis. These artificial neural networks (also called convolutional neural networks, or CNNs) are capable of categorizing information from the simplest to the most complex. Convolutional networks are characterized by two parts of layers, the convolutional part, and the classification part. Where the convolutional part consists of

extracting the features of input images based on convolutional mathematical operations, and the classification part consists of predicting the type and class of input images by computing the input images score of belonging to each class based on the information and characteristics provided by the previous part. To use a CNN, a learning phase is necessary on desired objects, but this phase is very costly in terms of data, time, and material resources. In order to overcome this problem, there are several open-source pre-trained models provided by several companies such as the Inception v3 model formed by Google. In the proposed method, we have used the transfer learning technique to use this pre-trained model, where it consists of 42 layers and recognizes almost 1000 types of objects.
We have chosen to use this model due to its high performance in terms of precision and also it does not consume many resources during the processing phase.

- *Candidate verification during night time*

For the purpose of classifying the connected components (lamp candidates) extracted in the previous step, we must first filter these candidates by their colours in the input image in order to keep only the white candidates, which represent the potential headlights, and the red candidates, which represent the potential taillights, and to omit all other components that are neither white nor red. Next, we generate all possible vehicle candidates by gathering all lamp candidates into pairs according to some specific criteria and consider the square image that bound this pair as vehicle candidate, where lamp candidates that are in the same horizontal position and fairly close to each other, and that have the same colour are considered as a pair. First, the potential lamps are grouped according to their colours. All potential lamps that have the same color are grouped together, red for potential tail lights and white for potential headlights. Then, they are grouped according to the size criteria in order to keep only the potential lamps that have approximately similar sizes. Next, they are filtered according to the position criteria, in order to group only the potential lamps that are horizontally aligned. And finally, they are grouped into sets according to the distance feature, in order to keep in each set only the potential lamps that are relatively close to each other. Note that a lamp candidate can be part of more than one pair.

Finally, we classify these generated vehicle candidates using the CNN classifier, where it is retrained to classify them as vehicles or non-vehicles. The dataset used to retrain the CNN contains positive samples, which are vehicle images collected from several videos from several countries, and negative samples, which are various images containing different objects or parts of the road.

3. Experiment results

3.1. Performance metrics

In purpose to validate the proposed system, five video sequences are recorded in different environments with different conditions. Three video sequences are recorded during day-time and the other two videos are recorded during night-time. The system proposed was evaluated and executed using C++, OpenCV and TensorFlow on a dual-core ARM Cortex-A9 processor with a clock speed of 1.2 GHz, running on an LXDE desktop with 1 GB DDR3 memory. This processor is integrated in a VEEK-MT2S, which is made up of a DE10 standard FPGA and a MTLC2 module supplied by TERASIC.

We have relied on the precision and recall metrics in order to show the overall accuracy, these metrics are calculated as follows:

$$Precision\ Rate = \frac{TP}{TP+FP}. \quad (1)$$

$$Recall\ Rate = \frac{TP}{TP+FN}, \quad (2)$$

where TP is true positives, FP is false positives and FN is false negatives.

3.2. Evaluation results

In order to correctly assess the proposed system, we have utilized four works from the literature in addition to our previous work to compare our system with, two works are used to assess the vehicle detection method during day and the other two works are used to assess the vehicle detection method during night. The comparison results listed in Table 1 shows that the proposed method has the highest accuracy and that the improvement done to the proposed method have boosted the accuracy. Some results of the detection during day are shown in Figure 6. The comparison results listed in Table 2 shows that the proposed

method achieves high accuracy and efficiently detects vehicles during night. Some results of the detection during night are shown in Figure 7.

Table 1. Evaluation results of the four vehicle detection methods during day

Videos	**1**		**2**		**3**	
Methods	**Precision %**	**Recall %**	**Precision %**	**Recall %**	**Precision %**	**Recall %**
(Robert *et al.*, 2017)	95.95	93.68	99.05	92.63	98.21	91.29
(Yong *et al.*, 2017)	97.99	96.44	98.51	97.64	97.51	97.51
Our previous work (Slimani *et al.*, 2018)	93.15	90.05	91.09	91	92.27	91.03
Our proposed work	98.05	99.60	98.83	100	97.96	99.59

Table 2. Evaluation results of the three vehicle detection methods during night

Videos	**1**		**2**	
Methods	**Precision (%)**	**Recall (%)**	**Precision (%)**	**Recall (%)**
(Gao *et al.*, 2019)	96.92	96.92	97.26	96.68
Vu *et al.*, 2017)	96.44	95.59	96.96	96.37
Our proposed work	98.24	98.24	98.78	98.49

Moreover, according to the information presented in Figure 5, our method is not only efficient in terms of detection, but also fast and can operate in real time. Figure 4 shows that the daytime detection method can process up to 23 frames per second in the three daytime videos, and the night-time detection method can process up to 23.22 frames per second in the two night-time videos.

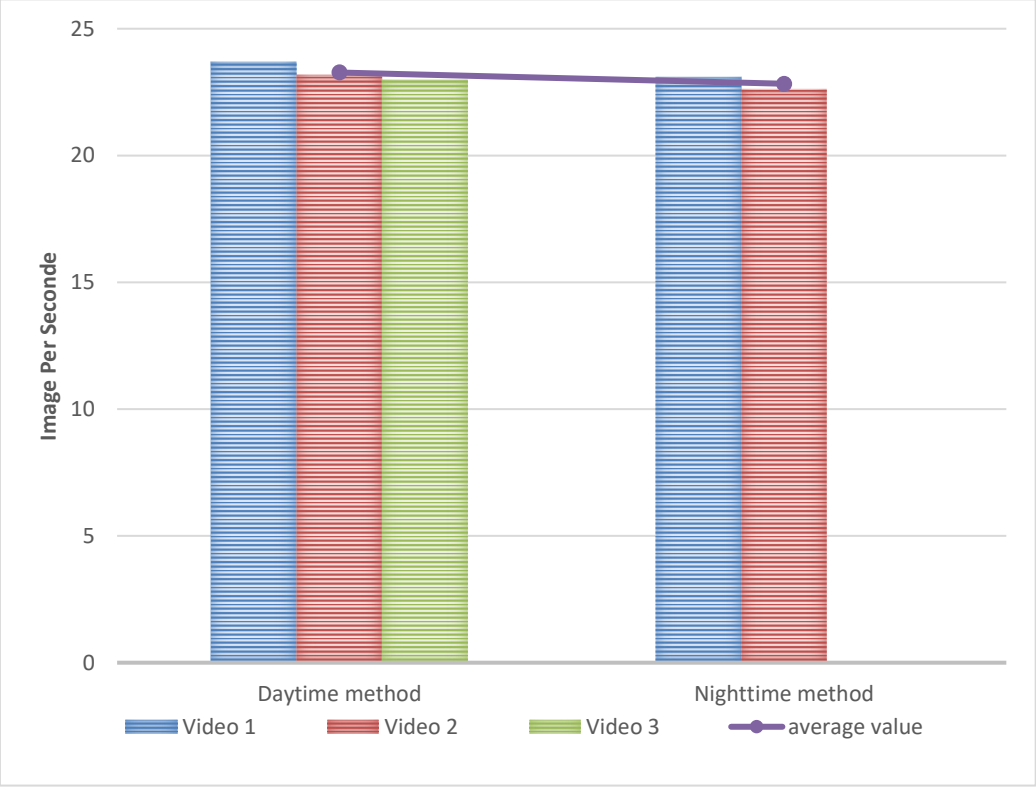


Figure 5. Number of images per second processed by each method

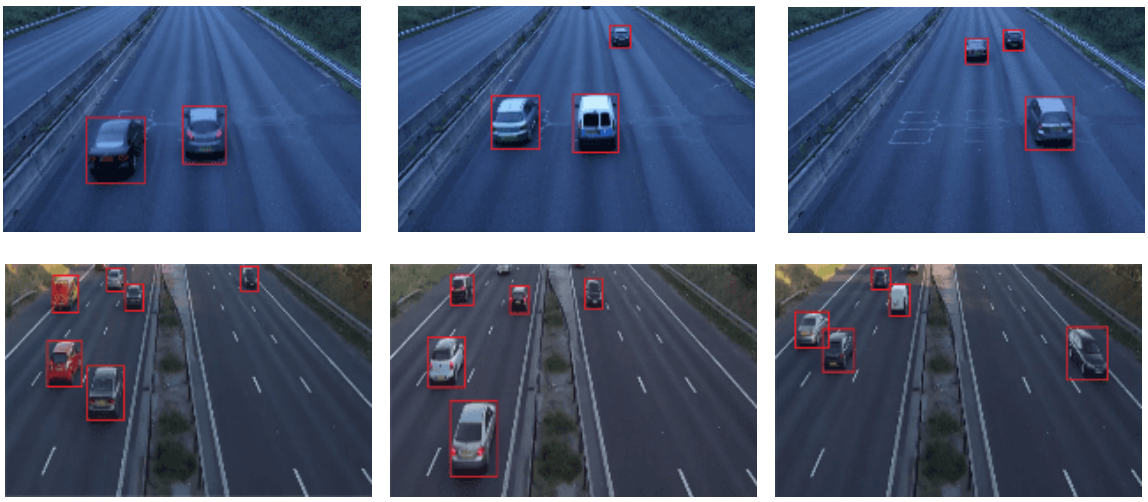


Figure 6. Some results of the vehicle detection method during day

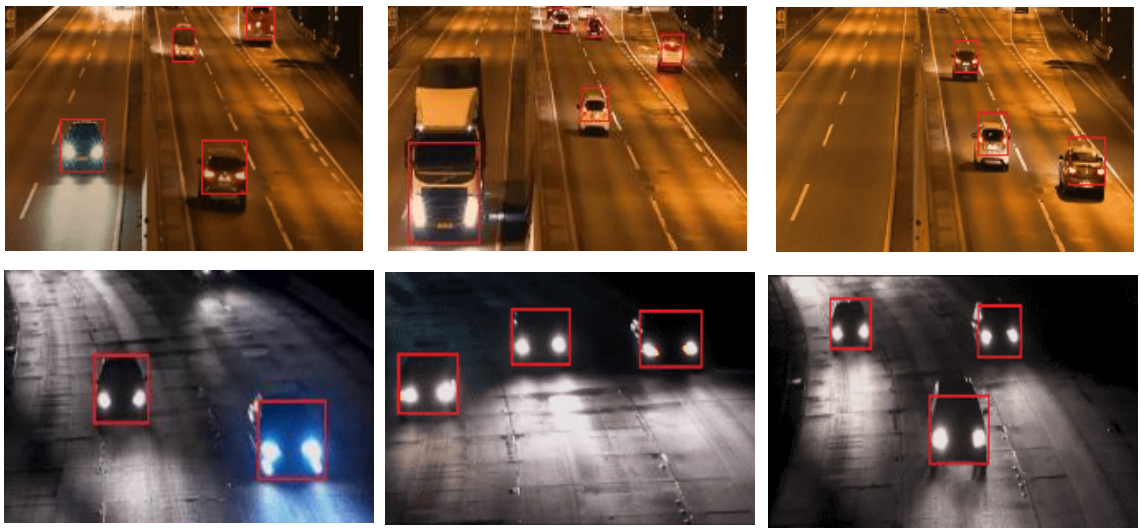


Figure 7. Some results of the vehicle detection method during night

4. Conclusions

A vehicle detection system for traffic monitoring is proposed in this paper. It addresses day and night vehicle detection using 2D-DWT and CNN classifier. The proposed system is composed of two detection modes: day detection mode and night detection mode. Both modes are divided into two main steps: the candidate generation step and the candidate verification step. In the first step of both detection modes, we generate the candidates by first performing the second level of 2D-DWT. Then, we produce a binary image based on background subtraction using two different thresholds depending on the detection mode used (day detection mode or night detection mode), and to improve the background subtraction results, we periodically update the background image. Then, we recover the missing pixels due to the binarization problem produced in the binary image, using morphological operations. Then, we extract each generated candidate (vehicle candidate/lamp candidate) as a connected component by applying the connected component technique. In the second step of the first detection mode, we verify and classify directly the generated candidates using the CNN classifier. In the second step of the second detection mode, the generated candidates are colour-filtered in order to preserve the potential headlamps and rear-lamps. Then, we generate all the possible pairs according to their distance, their colour and their horizontal level. Then, the generated pairs are classified using the CNN classifier. The proposed method has been tested in experiments and has worked well in terms of accuracy and precision. Moreover, the comparison made between the proposed system and other systems in the literature showed that the proposed system reaches the highest accuracy.

Data availability

The data used to support the findings of this study are available from the corresponding author upon request.

Declaration of competing interest

The authors declare that there are no conflicts of interests exist.

References

1. Al Okaishi, W., Zaarane, A., Slimani, I., Atouf, I. (2019) A Traffic Surveillance System in Real-Time to Detect and Classify Vehicles by Using Convolutional Neural Network. In: *2019 International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS)*, 1-5. IEEE.
2. Bell, A., Mantecón, T., Díaz, C., del-Blanco, C., Jaureguizar, F., García, N. (2021) A novel system for nighttime vehicle detection based on foveal classifiers with real-time performance. *IEEE Transactions on Intelligent Transportation Systems*, 23(6), 5421-5433.
3. Billones, R. K., Bandala, A., Sybingco, E., Gan Lim, L. (2017) Vehicle detection and tracking using corner feature points and artificial neural networks for a vision-based contactless apprehension system. In: *2017 Computing Conference*, 688-691. IEEE.
4. Gao, F., Ge, Y., Lu, S., Zhang, Y. (2019) On-line vehicle detection at nighttime-based tail-light pairing with saliency detection in the multi-lane intersection. *IET Intelligent Transport Systems*, 13(3), 515-522.
5. Gonzalez, R.C., Woods, R.E. (1992) Digital Image Processing. Addison Wesley, Reading, MA.
6. Nogueira K., Penatti, O., dos Santos, J. A. (2017) Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 61, 539–556.
7. Nurhadiyatna, A., Jatmiko, W., Hardjono, B., Wibisono, A., Sina, I., Mursanto, P. (2013) Background Subtraction Using Gaussian Mixture Model Enhanced by Hole Filling Algorithm. In*: Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics*, 4006-4011.
8. Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016) You Only Look Once: Unified, Real-Time Object Detection. In: *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788. Seattle, WA, USA, 27–30 June 2016.
9. Slimani, I., Zaarane, A., Al Okaishi, W., Atouf, I., Hamdoun, A. (2020) An Automated License Plate Detection and Recognition System Based on Wavelet Decomposition and CNN. *Array*, 2020, p. 100040.
10. Slimani, I., Zaarane, A., Hamdoun, A. (2016) Convolution algorithm for implementing 2D discrete wavelet transform on the FPGA. In: *IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, 1-3. IEEE.
11. Slimani, I., Zaarane, A., Hamdoun, A., Atouf, I. (2018) Traffic surveillance system for vehicle detection using discrete wavelet transform. *J Theor Appl Inf Technol*, 96(17), 5905–17.
12. Slimani, I., Zaarane, A., Hamdoun, A., Atouf, I. (2019) Vehicle License Plate Localization and Recognition System for Intelligent Transportation Applications. In: *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, 1592-1597. IEEE, 2019.
13. Sudha, D., Priyadarshini, J. (2020) An intelligent multiple vehicle detection and tracking using modified vibe algorithm and deep learning algorithm. *Soft Computing*, 24(22), 17417-17429.
14. Tang, Y., Zhang, C., Gu, R., Li, P., Yang, B. (2017) Vehicle detection and recognition for intelligent traffic surveillance system. *Multimedia tools and Applications*, 76(4), 5817-5832.
15. Velazquez-Pupo, R., Sierra-Romero, Al., Torres-Roman, D., Shkvarko, Y. (2018) Vehicle detection with occlusion handling, tracking, and OC-SVM classification: A high performance vision-based system. *Sensors*, 18(2), 374.
16. Vu, T.A., Pham, L. H., Huynh, T. K., Ha, S. (2017) Nighttime vehicle detection and classification via headlights trajectories matching. In: *2017 International Conference on System Science and Engineering (ICSSE)*, 221-225. IEEE.
17. Miao, Y., Liu, F., Hou, T., Liu, L. (2020) A nighttime vehicle detection method based on YOLO v3. In: *Chinese Automation Congress (CAC)*, 6617-6621. IEEE.
18. Zaarane, A., Slimani, I., Al Okaishi, W., Atouf, I., Hamdoun, A. (2020) Distance measurement system for autonomous vehicles using stereo camera. *Array*, 5, 100016.

19. Zaarane, A., Slimani, I., Hamdoun, A., Atouf, I. (2019b) Vehicle to vehicle distance measurement for self-driving systems. In: *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, 1587-1591. IEEE.
20. Zaarane, A., Slimani, I., Hamdoun, A., Atouf, I. (2019a) Real-Time Vehicle Detection Using Cross-Correlation and 2D-DWT for Feature Extraction. *Journal of Electrical and Computer Engineering*, 2019.
21. Zhang, Y., Guo, Z., Wu, J., Tian, Y. (2022) Real-Time Vehicle Detection Based on Improved YOLO v5. *Sustainability*, 14(19), 12274.